SPYWOLF

Security Audit Report

(TESTNET)

Completed on
March 17, 2023

SPYWOLF.CO
@SPYWOLFNETWORK
@SPYWOLFNETWORK

OVERVIEW

This audit has been prepared for **VV Token** to review the main aspects of the project to help investors make make an informative decision during their research process.

You will find a a summarized review of the following key points:

- ✓ Contract's source code
- ✓ Owners' wallets
- ✓ Tokenomics
- ✓ Team transparency and goals
- ✓ Website's age, code, security and UX
- ✓ Whitepaper and roadmap
- ✓ Social media & online presence

> *The results of this audit are purely based on the team's evaluation and does not guarantee nor reflect the projects outcome and goal*
>
> - SPYWOLF Team -

TABLE OF CONTENTS

VV Token

PROJECT DESCRIPTION

According to their website:

$VV Token is the engine powering the VV ecosystem. $VV token is the easiest device for using every feature VV has to offer including:

- Access to virtual entertainment experiences like concerts, museum exhibitions, and comedy shows in Unus World
- Discounts on wearable NFTs for avatars
- First access to the 6-District VV Metaverse
- Purchasing, rent, and building land

Release Date: Presale starts in March, 2023

Category: Metaverse

CONTRACT INFO

Token Name ToklenVesting	Symbol N/A
Contract Address 0x4247A9FA0973bdeCB8E65f5E205e4BF30F999fDB	
Network Ethereum **Goerli TESTNET**	Language Solidity
Deployment Date March 15, 2023	Verified? Yes
Total Supply N/A	Status Deployed

TAXES

Buy Tax **none**	Sell Tax **none**

Our Contract Review Process

The contract review process pays special attention to the following:

- ✓ Testing the smart contracts against both common and uncommon vulnerabilities
- ✓ Assessing the codebase to ensure compliance with current best practices and industry standards.
- ✓ Ensuring contract logic meets the specifications and intentions of the client.
- ✓ Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- ✓ Thorough line-by-line manual review of the entire codebase by industry experts.

Blockchain security tools used:

- OpenZeppelin
- Mythril
- Solidity Compiler
- Hardhat

CURRENT STATS

(As of March 16, 2023)

Status:

Not Launched!

Liquidity

Not added yet

Burn

No burnt tokens

MaxTxAmount

No limit

LP Address(es)

Liquidity not added yet

TOKEN TRANSFERS STATS

Transfer Count	TESTNET
Uniq Senders	TESTNET
Uniq Receivers	TESTNET
Total Amount	TESTNET
Median Transfer Amount	TESTNET
Average Transfer Amount	TESTNET
First transfer date	TESTNET
Last transfer date	TESTNET
Days token transferred	TESTNET

SMART CONTRACT STATS

Calls Count	TESTNET
External calls	TESTNET
Internal calls	TESTNET
Transactions count	TESTNET
Uniq Callers	TESTNET
Days contract called	TESTNET
Last transaction time	TESTNET
Created	TESTNET
Create TX	TESTNET
Creator	TESTNET

FEATURED WALLETS

Owner address	0xB5359AfCe552240C6EF3c48C321A40EF21DEffaB
LP address	**N/A**

TOP 3 UNLOCKED WALLETS

1 N/A

2 N/A

3 N/A

VULNERABILITY CHECK

Design Logic	Passed
Compiler warnings.	Passed
Private user data leaks	Passed
Timestamp dependence	Passed
Integer overflow and underflow	Passed
Race conditions and reentrancy. Cross-function race conditions	Passed
Possible delays in data delivery	Passed
Oracle calls	Passed
Front running	Passed
DoS with Revert	Passed
DoS with block gas limit	Passed
Methods execution permissions	Passed
Economy model	Passed
Impact of the exchange rate on the logic	Passed
Malicious Event log	Passed
Scoping and declarations	Passed
Uninitialized storage pointers	Passed
Arithmetic accuracy	Passed
Cross-function race conditions	Passed
Safe Zeppelin module	Passed
Fallback function security	Passed

THREAT LEVELS

When performing smart contract audits, our specialists look for known vulnerabilities as well as logical and access control issues within the code. The exploitation of these issues by malicious actors may cause serious financial damage to projects that failed to get an audit in time. We categorize these vulnerabilities by the following levels:

High Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

Medium Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

Low Risk

Issues on this level are minor details and warning that can remain unfixed.

Informational

Information level is to offer suggestions for improvement of efficacy or security for features with a risk free factor.

FOUND THREATS

High Risk

No high risk-level threats found in this contract.

Medium Risk

No medium risk-level threats found in this contract.

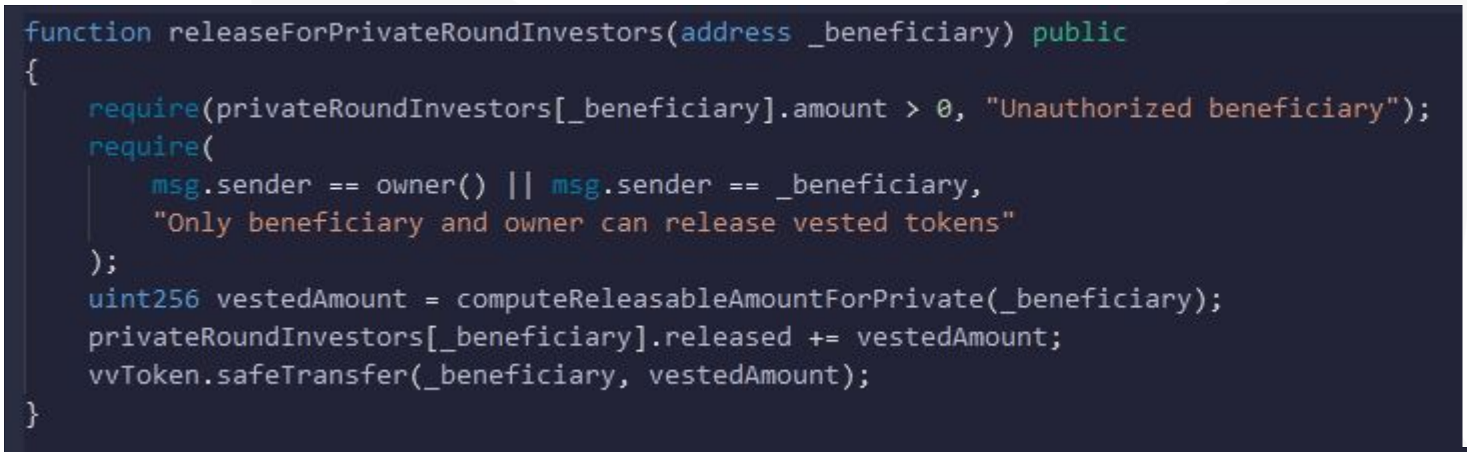
Low Risk

No low risk-level threats found in this contract.

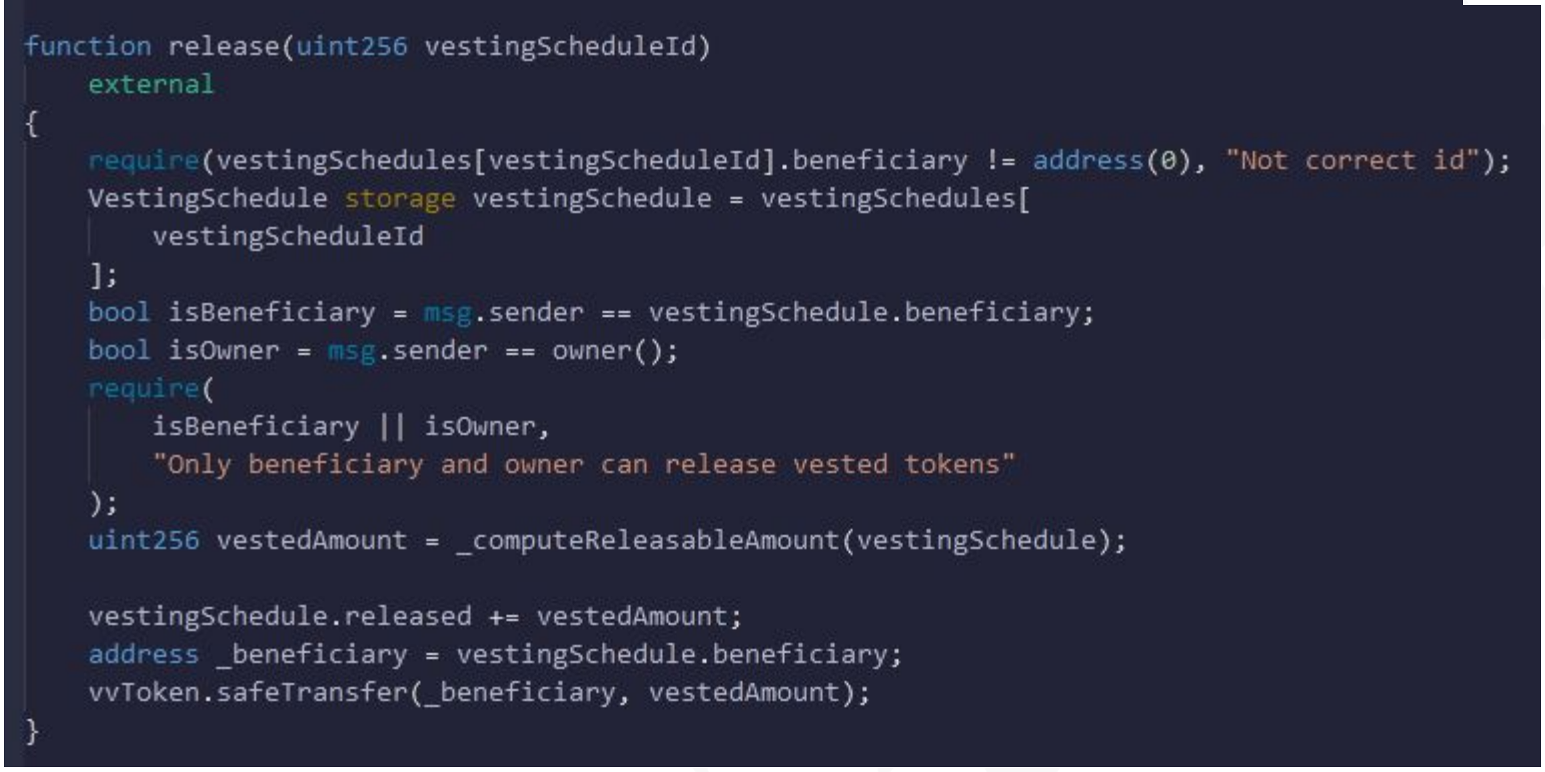
FOUND THREATS

⚠ High Risk

Release tokens vesting functions do not check and compare total unreleased amounts vs total amount available for release.
This can lead to contract drain of VV Tokens if functions are called in repetitive manner.



```
function releaseForPrivateRoundInvestors(address _beneficiary) public
{
    require(privateRoundInvestors[_beneficiary].amount > 0, "Unauthorized beneficiary");
    require(
        msg.sender == owner() || msg.sender == _beneficiary,
        "Only beneficiary and owner can release vested tokens"
    );
    uint256 vestedAmount = computeReleasableAmountForPrivate(_beneficiary);
    privateRoundInvestors[_beneficiary].released += vestedAmount;
    vvToken.safeTransfer(_beneficiary, vestedAmount);
}
```



```
function release(uint256 vestingScheduleId)
    external
{
    require(vestingSchedules[vestingScheduleId].beneficiary != address(0), "Not correct id");
    VestingSchedule storage vestingSchedule = vestingSchedules[
        vestingScheduleId
    ];
    bool isBeneficiary = msg.sender == vestingSchedule.beneficiary;
    bool isOwner = msg.sender == owner();
    require(
        isBeneficiary || isOwner,
        "Only beneficiary and owner can release vested tokens"
    );
    uint256 vestedAmount = _computeReleasableAmount(vestingSchedule);

    vestingSchedule.released += vestedAmount;
    address _beneficiary = vestingSchedule.beneficiary;
    vvToken.safeTransfer(_beneficiary, vestedAmount);
}
```

Recommendation: Ensure that check of currently distributed tokens and total available tokens for release is performed with each function call. Total released tokens should not exceed the amount variable set in every struct.

Informational

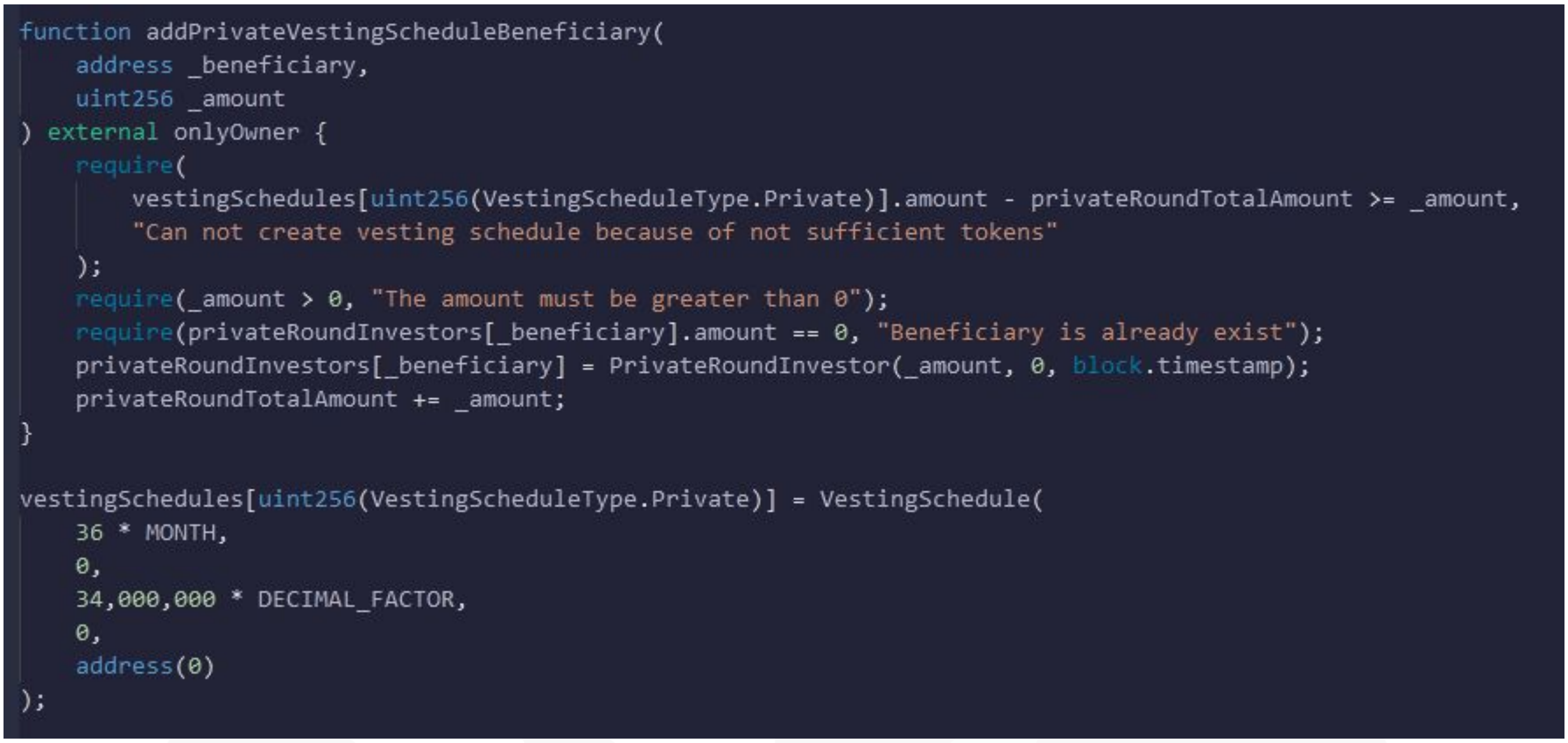
Token vesting schedules are as follows:

Seed round investors - 48 Months 20,000,000 tokens, no tokens released immediately.
Public round investors - 9 Months perdiod total of 40,000,000 tokens, 4,000,000 (10%) released immediately.
OperationsAndReserve vesting - 48 months period for total of 456,000,000 tokens, 45,600,000 (10%) released immediately.
SocialAdvisory vesting - 12 Months period for total of 60,000,000 tokens, no tokens released immediately.
LiquidityAndListings vesting - 48 Months period for total of 100,000,000 tokens, 5,000,000 (5%) released immediately.
Founders vesting - 48 Monts period for total of 220,000,000 tokens, no tokens released immediately.
Private round vesting - 36 Months period for total of 34,000,000 tokens, no tokens released immediately.
MarketingAndTechDevelopment - No vesting period, 70,000,000 tokens released immediately.

Only addresses added in the private round can withdraw tokens directly from the contract.
Tokens vested in different schedules than the private one will be sent to the corresponding beneficiary addresses.
Only owner can the corresponding beneficiary addresses can withdraw tokens from the said vesting schedules.

Informational

Owner can add addresses and assign share in the private round sale, until total combined amount of added users reach 34,000,000 tokens.

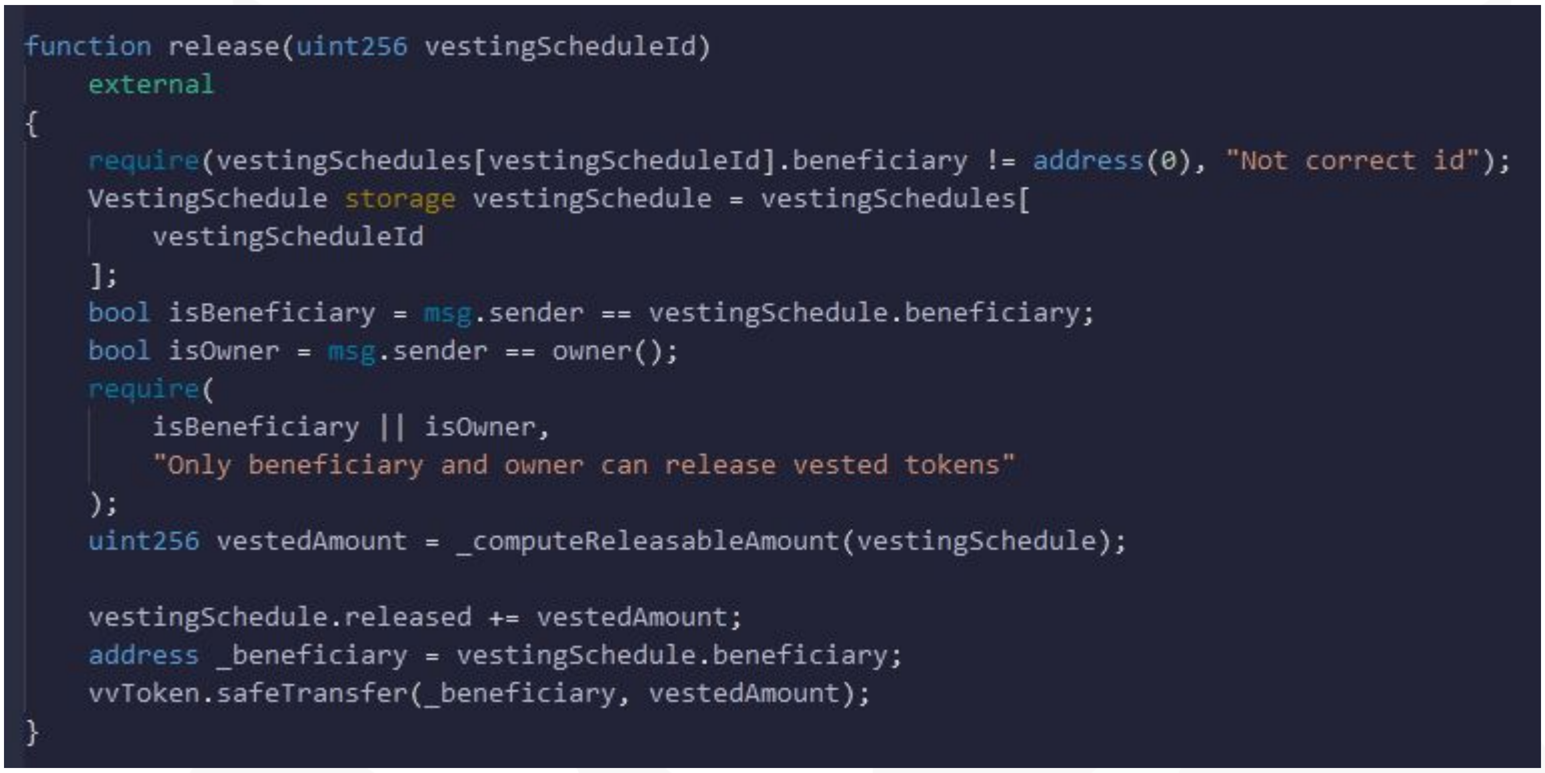


```
function addPrivateVestingScheduleBeneficiary(
    address _beneficiary,
    uint256 _amount
) external onlyOwner {
    require(
        vestingSchedules[uint256(VestingScheduleType.Private)].amount - privateRoundTotalAmount >= _amount,
        "Can not create vesting schedule because of not sufficient tokens"
    );
    require(_amount > 0, "The amount must be greater than 0");
    require(privateRoundInvestors[_beneficiary].amount == 0, "Beneficiary is already exist");
    privateRoundInvestors[_beneficiary] = PrivateRoundInvestor(_amount, 0, block.timestamp);
    privateRoundTotalAmount += _amount;
}

vestingSchedules[uint256(VestingScheduleType.Private)] = VestingSchedule(
    36 * MONTH,
    0,
    34,000,000 * DECIMAL_FACTOR,
    0,
    address(0)
);
```

Informational

Owner can release any vesting schedule to their corresponding beneficiary address.
Beneficiary address for the current vesting schedule can also initiate the function.

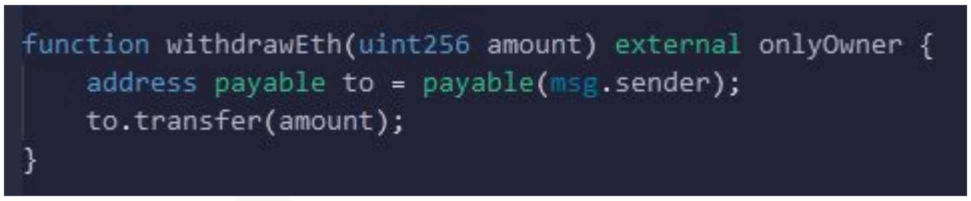


```
function release(uint256 vestingScheduleId)
    external
{
    require(vestingSchedules[vestingScheduleId].beneficiary != address(0), "Not correct id");
    VestingSchedule storage vestingSchedule = vestingSchedules[
        vestingScheduleId
    ];
    bool isBeneficiary = msg.sender == vestingSchedule.beneficiary;
    bool isOwner = msg.sender == owner();
    require(
        isBeneficiary || isOwner,
        "Only beneficiary and owner can release vested tokens"
    );
    uint256 vestedAmount = _computeReleasableAmount(vestingSchedule);

    vestingSchedule.released += vestedAmount;
    address _beneficiary = vestingSchedule.beneficiary;
    vvToken.safeTransfer(_beneficiary, vestedAmount);
}
```

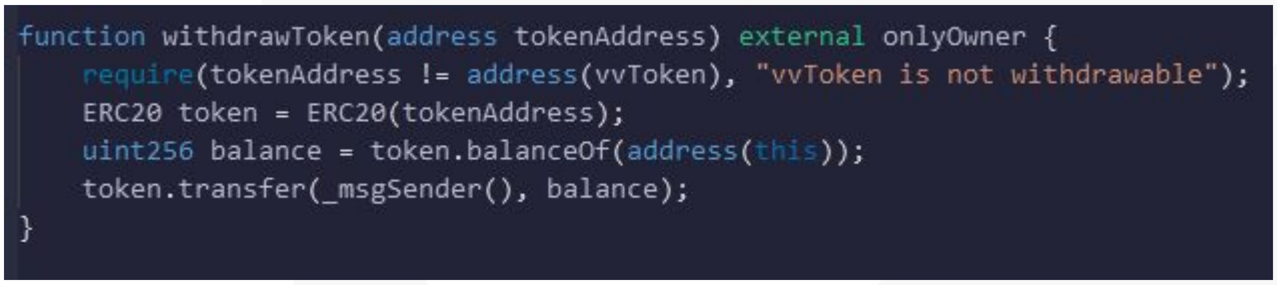
Informational

Owner can withdraw ETH from the contract.



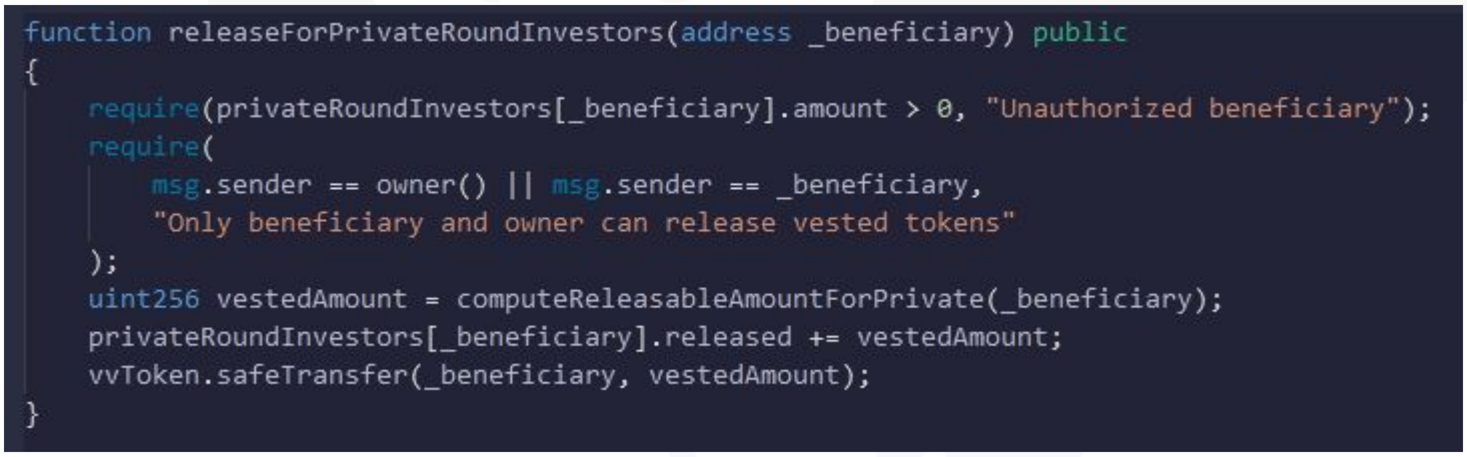
```
function withdrawEth(uint256 amount) external onlyOwner {
    address payable to = payable(msg.sender);
    to.transfer(amount);
}
```

Owner can withdraw any tokens from the contract with exception for the VV Token.



```
function withdrawToken(address tokenAddress) external onlyOwner {
    require(tokenAddress != address(vvToken), "vvToken is not withdrawable");
    ERC20 token = ERC20(tokenAddress);
    uint256 balance = token.balanceOf(address(this));
    token.transfer(_msgSender(), balance);
}
```

Owner can release private investors vesting schedule to the corresponding beneficiary address.
Private investors can also initiate that function if they are the beneficiary address.



```
function releaseForPrivateRoundInvestors(address _beneficiary) public
{
    require(privateRoundInvestors[_beneficiary].amount > 0, "Unauthorized beneficiary");
    require(
        msg.sender == owner() || msg.sender == _beneficiary,
        "Only beneficiary and owner can release vested tokens"
    );
    uint256 vestedAmount = computeReleasableAmountForPrivate(_beneficiary);
    privateRoundInvestors[_beneficiary].released += vestedAmount;
    vvToken.safeTransfer(_beneficiary, vestedAmount);
}
```

RECOMMENDATIONS FOR

GOOD PRACTICES

1. Consider fundamental tradeoffs
2. Be attentive to blockchain properties
3. Ensure careful rollouts
4. Keep contracts simple
5. Stay up to date and track development

TokenVesting

GOOD PRACTICES FOUND

✓ The owner cannot stop or pause the contract after start.

CONTRACT INFO

Token Name
Virtual Versions

Symbol
VV

Contract Address
0xE200a7Cb66EB18ca9BCF1806DB6a882f025DaE0A

Network
Ethereum **Goerli TESTNET**

Language
Solidity

Deployment Date
March 15, 2023

Verified?
Yes

Total Supply
1,000,000,000

Status
Deployed

TAXES

Buy Tax
none

Sell Tax
none

Our Contract Review Process

The contract review process pays special attention to the following:

- ✓ Testing the smart contracts against both common and uncommon vulnerabilities
- ✓ Assessing the codebase to ensure compliance with current best practices and industry standards.
- ✓ Ensuring contract logic meets the specifications and intentions of the client.
- ✓ Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- ✓ Thorough line-by-line manual review of the entire codebase by industry experts.

Blockchain security tools used:

- OpenZeppelin
- Mythril
- Solidity Compiler
- Hardhat

FOUND THREATS

High Risk

No high risk-level threats found in this contract.

Medium Risk

No medium risk-level threats found in this contract.

Low Risk

No low risk-level threats found in this contract.

RECOMMENDATIONS FOR

GOOD PRACTICES

1. Consider fundamental tradeoffs
2. Be attentive to blockchain properties
3. Ensure careful rollouts
4. Keep contracts simple
5. Stay up to date and track development

VV Token

GOOD PRACTICES FOUND

- ✓ The owner cannot mint new tokens after deployment
- ✓ The owner cannot stop or pause the contract
- ✓ The owner cannot set a transaction limit

SPYWOLF

CRYPTO SECURITY

Audits | KYCs | dApps
Contract Development

ABOUT US

We are a growing crypto security agency offering audits, KYCs and consulting services for some of the top names in the crypto industry.

- ✓ **OVER 150 SUCCESSFUL CLIENTS**
- ✓ **MORE THAN 500 SCAMS EXPOSED**
- ✓ **MILLIONS SAVED IN POTENTIAL FRAUD**
- ✓ **PARTNERSHIPS WITH TOP LAUNCHPADS, INFLUENCERS AND CRYPTO PROJECTS**
- ✓ **CONSTANTLY BUILDING TOOLS TO HELP INVESTORS DO BETTER RESEARCH**

To hire us, reach out to
contact@spywolf.co or
t.me/joe_SpyWolf

FIND US ONLINE

SPYWOLF.CO	SPYWOLF.NETWORK
@SPYWOLFNETWORK	@SPYWOLFOFFICIAL
@SPYWOLFNETWORK	@SPYWOLFNETWORK

Disclaimer

This report shows findings based on our limited project analysis, following good industry practice from the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, overall social media and website presence and team transparency details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report.

While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – please make sure to read it in full.

DISCLAIMER:

By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice.

No one shall have any right to rely on the report or its contents, and SpyWolf and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (SpyWolf) owe no duty of care towards you or any other person, nor does SpyWolf make any warranty or representation to any person on the accuracy or completeness of the report.

The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and SpyWolf hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, SpyWolf hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against SpyWolf, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report. The analysis of the security is purely based on the smart contracts, website, social media and team.

No applications were reviewed for security. No product code has been reviewed.